**Noname manuscript No.**
(will be inserted by the editor)

# Data Quality Assessment in Digital Score Libraries

## The GioQoso Project

**Francesco Foscarin · Philippe Rigaux · Virginie Thion**

Received: date / Accepted: date

**Abstract** *Sheet music scores* have been the traditional way to preserve and disseminate western classical music works for centuries. Nowadays, their content can be encoded in digital formats that yield a very detailed representation of music content expressed in the language of *music notation*. These *digital scores* constitute, therefore, an invaluable asset for digital library services such as search, analysis, clustering, recommendations, and synchronization with audio files.

Digital scores, like any other published data, may suffer from quality problems. For instance, they can contain incomplete or inaccurate elements. As a "dirty" dataset may be an irrelevant input for some use cases, users need to be able to estimate the quality level of the data they are about to use.

This article presents the data quality management framework for digital score libraries (DSL) designed by the GioQoso multidisciplinary project. It relies on a *content model* that identifies several information levels that are unfortunately blurred out in digital score encodings. This content model then serves as a foundation to organize the categories of quality issues that can occur in a music score, leading to a *quality model*. The quality model also positions each issue with respect to potential usage contexts, allowing attachment of a consistent set of indicators that together measure how a given score is *fit* to a specific usage. We finally report an implementation of these conceptual foundations in an online DSL.

Francesco Foscarin, Philippe Rigaux
CEDRIC, CNAM
Paris, France
E-mail: firstname.lastname@cnam.fr

Virginie Thion
Univ Rennes, CNRS, IRISA
Lannion, France
E-mail: firstname.lastname@irisa.fr

## 1 Introduction/motivation

Music is an essential part of the world's cultural heritage. Even though recordings and audio files constitute the main access channel to music works nowadays, music has been preserved and disseminated as *sheet scores* for centuries. For a part of music production, sheet scores have been - and continue to be - the most complete and accurate way to encode the composer's intents, and faithfully convey these intents to performers.

A sheet score is a complex semiotic object. In a single and compact layout, it combines a symbolic encoding of the music that must be produced with a sophisticated visual representation that aims at accurately representing the music content.

As an illustration, Figure 1 shows an excerpt of Cherubini's *Dies Irae*[1] available online on the Neuma platform [23] (we discuss Neuma's architecture further in Section 5). It contains four parts that are intended to be played synchronously, with each one assigned to a performer (a singer). Each part consists of a single staff, which starts with a key and time signature, followed by seven measures (separated by vertical bars). Each measure contains notes: their vertical position describes frequency information and their shape encodes the temporal duration. Lyrics, expressed by a sequence of syllables, are associated with the notes. Other signs,

[1] The complete digital score is available at `http://neuma.huma-num.fr/home/opus/composers:cherubini:Requiem_Cherubini_Dies_Ireae/`.

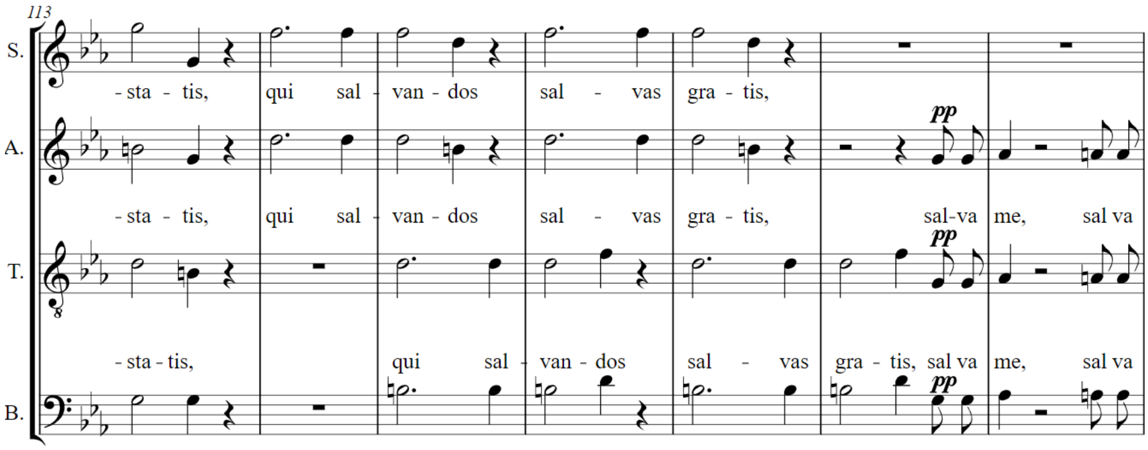


**Fig. 1** Music score extracted from the NEUMA Platform [27]

such as the *pianissimo* annotation in measure 6, carry information on *how* the notes have to be played.

Sheet music scores are able to represent music that involve dozens of performers (e.g., a large orchestra and a choir) at a very detailed level. Rich musical notation is a key asset for encoding such a complex set of information. An obvious consequence, though, is that high-quality scores are difficult and costly to produce, which probably explains why they are considerably less represented than mere textual content in the mass of digital documents.

### 1.1 From images to music notation encoding

The largest and probably most used online collection of scores is the *International Music Score Library Project* (IMSLP) [19]. At the time of writing, it has published about 132,000 music works encoded in various formats. For the most part, IMSLP documents are simply scans of public domain scores, such as original engravings of early music works, or more recent publications that are nevertheless old enough to be copyright free. IMSLP is an important asset for musicians who can instantly access rare music pieces that would have required, a few years ago, a costly and time-consuming journey to some distant library. However, the nature of these documents, in most cases PDF images, has no benefit beyond their improved accessibility. They remain restricted to traditional usages as a support for either performance or visual, human-based analysis. This statement can be generalized to other large libraries that give online access to their collections, mostly in the form of images with generic navigation tools, and remain agnostic of the represented content. A representative example is the Gallica [1] digital library managed by the *Bibliothèque nationale de France*, which collects, records, and promotes the French documentary heritage.

In general, we expect a digital library to be more than a simple repository of digital documents, encoded in a neutral format that obscures their content. Services that leverage digital representation are required, and at the very least a search engine that allows retrieval of documents that match patterns of interest *by content*. However, even this simple requirement cannot be met in a score library based on PDF scans. The same holds for the many more specialized services that could be envisaged, specifically: frequent patterns and features (themes, cadences, harmonic progressions), extraction, audio rendering of music and music fragments, alignment with other sources (such as a digitized manuscript), collaborative annotations and editing, *etc.*

To supply such intelligent services, we need a digital representation that truly encodes the *music notation* embedded in a sheet score, and thereby gives fine-grained access to all its components. Such formats exist: MusicXML [17], MEI [29,21], or (in the near future) the MEX recommendation currently elaborated by the W3C Music Notation Community Group [22]. While the initial motivation was to address interoperability concerns, a side benefit of these structured encodings is access to the notation at a very detailed level. For instance, each note can be described with many attributes, including its pitch, octave, alteration, duration, positioning, stems, *etc.* This level of detail provides exciting perspectives regarding the emergence of new applications: automatic transformations (e.g., transpositions), synchronization with audio records, multimodal interactions (for instance, access for visually impaired people), and computer-aided analysis of the music itself.

From now on, the term *digital score* will denote any document that contains some piece of music described with the language of music notation, encoded in a computer-friendly language such as XML. *Digital score libraries* do exist in this restricted sense, but they remain quite limited in size with respect to, say, IMSLP. Often created and maintained by research institutes, they focus on a highly specialized repertoire, like *The Lost Voices Project* [20] designed by the *Centre d'Études Supérieures de la Renaissance* (CESR) of Tours (France), the NEUMA platform [23] designed by the *Conservatoire National des Arts et Métiers* (CNAM) and the *Institut de recherche en Musicologie* IReMus of Paris (France), or the *Global Chant Database* [16] designed by the Charles University of Prague (Czech Republic). The reason for this limited scope is clear: as already mentioned, digitizing music notation is quite a time-consuming task, and there is currently no institution or environment than can take on the enormous task of digitizing millions of scores. The OpenScore project [24] is a recent attempt that has interesting insights in its results, but it still exhibits limited production.

### 1.2 Massive digitization and quality assessment

Massive digitization processes are promising candidates to overcome the current bottleneck of human-driven score production. Several mechanisms can be envisioned: let us briefly discuss *music transcription* and *optical music recognition* (OMR). In the first case, the music notation is inferred from a human performance, typically encoded as a MIDI file. Although this constitutes an efficient input method compared to direct notation input, it still requires an expert user, and is essentially limited to keyboard works. OMR, on the other hand, takes the image of an existing sheet score and attempts to interpret its graphical components. In theory, the resulting digital notation could be as good as the input. But in practice, many mistakes are likely to occur. The rate of errors, as well as their seriousness, depends on many factors such as the graphical accuracy of the input, the number of instruments, the structure of the music work, and the complexity of the represented music itself (e.g., presence of sophisticated rhythmic and melodic patterns, non-standard notation signs, *etc.*). Tools that automatically detect and report quality problems over the resulting digital scores are needed to control the output of such digitization processes.

### 1.3 The case for an automatic quality assessment

In view of the preceding arguments, an automatic assessment of digital score quality, apt at investigating the core level of the notation and not the shallow graphical layout, is highly desirable. However, defining and measuring the quality of a score's encoding is not easy. Indeed, when one starts to figure out all the problems that can affect a score, they can quickly become quite overwhelming. Even worse, their apparent heterogeneity seems to prevent an organized approach. Some aspects are purely syntactic (do all slurs have a start/end point?), others pertain to metadata, which may or may not be mandatory (such as title, composer, date, or copyright). Some aspects are specific to the score layout (symbol overlapping, appropriate position of clefs and staves). And of course, the music content itself has to be correct regarding the source, and at least consistent with respect to editors' choices.

To avoid a case-by-case enumeration that would quickly result in a large, messy catalog, we need a more disciplined approach that identifies and structures the many facets of digital score production. We propose such an approach in the present paper.

The approach relies first on an original modeling of score material itself, which distinguishes the content, the semiotic artifacts used to represent this content in a so-called music score, and finally the specific rendering of these artifacts on a specific media, by a specific application. This preliminary modeling constitutes an essential part of the approach in our opinion, since it allows us to clearly identify, during the quality requirements specification step, the level at which these requirements operate. Moreover, this modeling is also the basis to determine the impact of a quality defect on a particular score usage. To give a simple example, putting a contrabasso and a piccolo part together on the same staff results in an awful visual representation, but remains harmless if the goal is an automatic harmonic analysis of the music content. We expose this *score model* in Section 2.

Based on the score model, we then embark on modeling the quality itself. Data quality turns out to be a complex, multi-dimensional concept. A traditional approach considers generic quality dimensions (completeness, accuracy, consistency) and enumerates requirements for each category. In the context of music scores, we consider that this "generic" approach is not satisfactory as it means enumerating a large set of quality requirements and indicators regardless of the complex process that mixes music content, performance directions, temporal synchronization, and readability concerns in this single complex artifact. We make the case for an approach that adopts the analysis of this score constitution process as a primary dimension, and to this end mobilizes the model elaborated in Section 2. We propose a taxonomy that identifies and characterizes the many facets that are interrelated in a score representation, and position the classical quality categorization at the level of these facets. This results in a two-dimensional quality model which seems the most appropriate to develop an informed and structured catalog of quality issues, and to further support a measurement of quality with respect to a specific usage. This contribution is presented in Section 3.

Equipped with this quality model, we address an essential aspect of data quality, namely *fitness for use*, as quality measurement involves dimensions and indicators that are relevant to a given user for a given *usage*. This means that a user $u_1$ may require some quality indicators for a specific usage, and some other indicators for another usage, which can be completely different than those needed by a user $u_2$. The *fitness for use* specificity of data quality is detailed, in the context of a DSL, in Section 4.

Finally, Section 5 covers the *implementation*, in an existing DSL, of the conceptual framework presented in the previous sections. We present the data quality module implemented in the NEUMA platform, which is an open repository of scores managed by both the CEDRIC and IReMus labs. We discuss architectural, data management, quality annotation, and quality visualization issues.

### 1.4 Positioning according to previous publications

The content of this article summarizes and enriches some of the scientific contributions produced during the GioQoso project [15] (2016-2019), funded by the French CNRS. This multidisciplinary project gathered musicologists and computer scientists[2] who have studied the problem of managing data quality in digital score libraries. Let us position the contributions that are presented in the following part of this paper on the basis of their previous contributions.

Section 2 presents the score model that the quality modeling is built on. It presents a part of the contributions proposed in [8] and [13], which define a content model relying on an ontology that exhaustively captures the relevant elements of a score's content. In Section 2, these contributions are summarized to the minimum needed to understand the data quality assessment process introduced after.

[2] The partners of the GioQoso project were the BnF – *Bibliothèque nationale de France* (Paris, France), the CEDRIC laboratory of the CNAM (Paris, France), the CESR – *Centre d'Études Supérieures de la Renaissance* (Tours, France), the iReMus – *Institut de recherche en Musicologie* (Paris, France) and the IRISA of Univ. Rennes (Lannion, France).

Section 3 and Section 4 concern the data quality model. They use the contributions presented in [6], [9], and [12] as a starting point, presented in the following over a unified point of view. We also enrich the presentation of the concepts with comprehensive use cases.

Finally, Section 5 addresses the implementation of the framework. In this section, we present the last version of the quality assessment tool that is based on the conceptual foundations introduced in the preceding sections. (A previous version of this tool was presented in [5].)

## 2 Modeling digital scores

This section covers the score production workflow model that clarifies the provenance, role, and significance of the information that can be found in digital score encodings. This model allows us to cope with the apparent heterogeneity of such content, and is an essential step to structure the quality requirements according to the production level at which they operate.

### 2.1 The music score production workflow

The workflow of (digital) score production [13] distinguishes the following three steps, as illustrated in Figure 2.

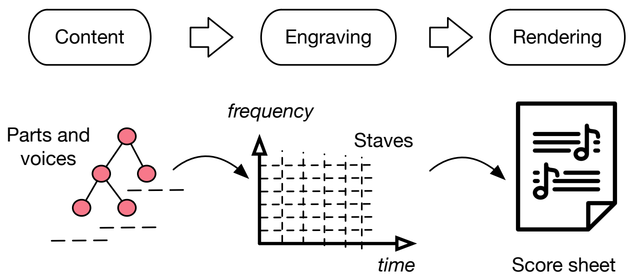


**Fig. 2** The workflow of (digital) score production

*(First step) The score content modeling.* This part covers all aspects related to what we call the *score content*, independently from any rendering concern. Essentially, it captures the *structural* organization of a score in parts and *streams* [7], and the description of streams as time-dependent elements.

*(Second step) The score engraving.* Score engraving is a set of instructions that details how score content has to be displayed on a media support. Music sound notation essentially specifies the production of a sound in a 2D space where time is the horizontal

axis and frequency is the vertical one. The engraving specification is thus modeled as a mapping that projects this score content in this 2D visualization space.

*(Third step) The score rendering.* The final step takes score content and score engraving specifications, and produces a layout of the score based on the properties of a specific media (paper, screen, *etc.*).

This workflow modeling is useful to identify and characterize the specific quality issues that can occur at each step, and to determine how we can evaluate and possibly fix these issues. An important first point is that the last step (score rendering) depends on the rendering software and on the properties of the displaying media. A high-quality score can be displayed very badly with a poor renderer or on a tiny screen. Therefore, we consider this part as out of scope for the quality evaluation process. This highlights the distinction between *score content* and *score engraving* quality issues. We consider that it makes sense for exactly the same reasons that led to separating the content of web pages (structured in HTML) from their display features (defined with CSS rules).[3]

This content modeling aims at capturing the part of music notation that abstracts the "true" music content explicitly found in a score encoding, and deliberately ignores issues related to the graphical rendering of scores. It requires the definition of the structure of a score.

On the other hand, *engraving* is a process that applies to the score content, and defines the relationships between this content and a two-dimensional space organized with respect to a temporal dimension (abscissa) and a frequency dimension (ordinate). Evaluating the engraving quality implies taking into account both the content and the mapping.

### 2.2 The data model

The data model is composed of three components: the score content model, the score engraving model, and the metadata.

#### *2.2.1 The score content model*

The "score content" part of the model [8] focuses on the aspects of a digital score representation that describes the intended production of sounds, and is independent from any visualization concern. More precisely, assume a "music rendering machine" $\mathcal{M}$ that takes a score $S$ as input and produces a music performance $P$ as output. We define the content of $S$ as the minimal subset of $S' \subseteq S$ such that $\mathcal{M}(\mathcal{S}') = \mathcal{M}(\mathcal{S}) = \mathcal{P}$.

This content definition depends on $\mathcal{M}$. A (too) simple candidate is a MIDI player, which sees a score, whatever its sophistication, as a piano roll and thus ignores score elements that are essential to a decent music performer, particularly meter and measures. On the other hand, any rendering machine (including the MIDI player) takes its input directly from the score encoding, and is not concerned by layout information designed to cope with the limitations of human readers. The allocation of music on staves and pages, for instance, is clearly not part of the score content in this respect. In general, in order to decide whether a piece of data belongs or not to the content, we just have to consider whether it is likely to influence the music production for an ideal performer with unhindered access to score information.

The score content is modeled as a *hierarchical structure*, where leaves consist of *voices*, and inner nodes of *parts*. Let us illustrate the structural aspect first with the sketch of a piano concerto score, depicted in Figure 3.

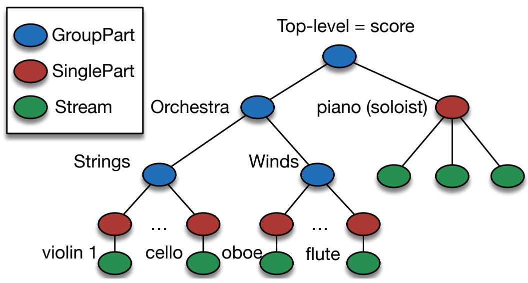


**Fig. 3** General structure of a score

The score is made of *parts*, where the concept of part is refined into two sub-concepts. A *group* (of parts) consists of a set of sub-parts, and mostly serves the organizational aspect of the score. For instance, the orchestral material of a concerto score typically defines a group for wind instruments, another one for string instruments, *etc.*

A *single part* encapsulates the music events assigned to an individual performer (instrument or vocal). As an example, Figure 3 shows a single part for the soloist (piano), and another one for the violins, cellos, *etc.* The information related to measures (in particular time signatures) are represented at this level. A single part contains one or several *voices*.

Finally, a *voice* is a timed series of *events*, where an event denotes the production of a sound artifact at a specific timestamp (the "onset"). Particular cases of

[3] The metaphor also holds for the *rendering step*, carried out in the case of HTML by a web browser that adjusts the textual content and CSS rules to the displaying window.

events are notes and chords (with pitch and duration information), textual contents, or information concerning dynamics and articulation.


**Fig. 4** Data model example: a score

*Example 1* Consider the score shown in Figure 4. It consists of two parts, let us call them “sopr” and “accompaniment”. The vocal part consists (in our modeling) of two voices: the first one (called “melody”) composed of sounds, and the second one (“lyrics”) of syllables (note that there is no one-to-one rhythmic correspondence between syllables and notes, as some syllables cover several notes). The second part consists of a single voice, “bass”.

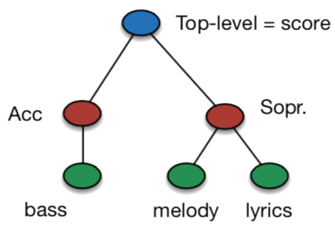


**Fig. 5** Data model example: structure of the score

The content structure corresponding to the score in Figure 4, in terms of the score content model that we introduced just before, is summarized in Figure 5. It details each voice. Voice “sopr” is a monophonic voice, with each event being either a single note or a rest. Voice “lyrics” is a sequence of syllabic events. Finally, voice “bass” contains a few complex events, with instances of chords.

The data model encompasses more advanced notions relying on an ontology that exhaustively models the content of a music score. The comprehensive ontology is presented in [8].

### *2.2.2 The score engraving model*

A score engraving embeds the musical content in a graphical representation that is structured according to the two following dimensions:

*Time.* This dimension is represented by the horizontal axis, and is discretized in measures, beats, and finite subdivisions of beats.

*Frequencies.* Sound frequencies are represented on a vertical axis, and are discretized in octaves, and subdivision of octaves in (usually) twelve semi-tones.

This yields a two-dimensional discretized space that could be represented as a grid. Each staff of a score can be fully displayed in this grid, each note being a segment whose height corresponds to its frequency, and length to the note duration. The score engraving model is based on this representation. This perspective is summarized in Figure 6. The engraving rules take a score's content, determine the number of staves, allocate parts to staves, and develop the stream representation on each staff.

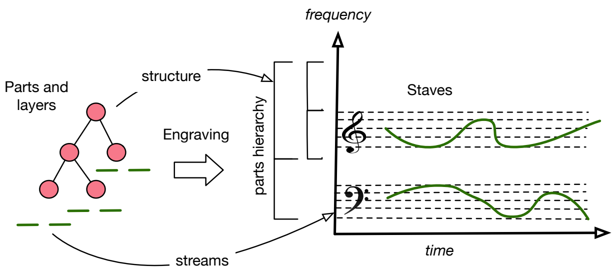


**Fig. 6** Engraving: mapping the content to (time, frequency) space

The quality model relies on this perspective, and focuses on the organization of staves, their relationships, and on the inner quality of stream representation for each staff. The general question that we try to address in this context is: to which extent does the content/staves mapping defined by the engraving ensure a consistent and correct layout of a score? If the engraving quality is high, then we can expect that a good renderer will be able to produce a readable score display at visualization time.

### *2.2.3 The metadata model*

Finally, we consider a third part of score encoding: its metadata. Metadata is data about data, *i.e.*, in our case, any content that annotates either the score content or the score engraving. The title, subtitle, and composer are metadata that annotate a score as whole. Instrument names annotate parts.

There are at least two reasons to incorporate metadata issues in quality evaluation. First, in some cases metadata supplies knowledge which is useful to measure a quality indicator. For example, knowing the instrument for a part makes it possible to check that the range of the music content is compatible with this instrument, or that the clef is appropriate. Second, metadata is typically a factor of inconsistencies when we consider quality concerns at a collection level. Music collection editors are eager to ensure that the level, accuracy, and encoding of metadata are similar for all the scores. Although our work focuses on independent scores, regardless of the collection where they are contained, this motivates the inclusion of metadata as part of our quality model.

## 3 Modeling data quality

The authors of [13,12] proposed a catalog of quality requirements specific to DSL data, which was elaborated from their experience in maintaining and using DSL. Based on the models introduced in the previous section, they defined a taxonomy for classifying these requirements. The taxonomy is like a forest, where each tree corresponds to a "facet" of quality evaluation, and contains the related set of requirements. The taxonomy contains three such trees, given in Figure 7. We partially describe it below. (The complete taxonomy is fully accessible at `http://neuma.huma-num.fr/quality`.)

- Score content issues
  - Structural issues
    - Unbalanced parts
  - Stream issues
    - Pitch
      - Out of range
    - Rhythm
      - Incomplete measures
      - Overflowing measures
    - Lyrics
      - Missing Lyrics
      - Invalid lyrics encoding
- Score engraving issues
  - Staves organization
    - Invalid staff order
    - Too many parts per staff
  - Staff parameters
    - Invalid key signature
    - Invalid clef
  - Staff layout
    - Erroneous Duration
    - Inappropriate beaming
- Metadata issues
  - Missing title
  - Missing composer
  - Invalid instrument name.
  - ...

**Fig. 7** Business functional taxonomy of quality requirements

Let us now detail the three main categories of quality issues considered in the quality model.

### 3.1 Score content issues

Score content issues concern either the structure of the content or the stream.

*Structural issues.* Some quality issues concern the structure of the score: the length of its elements and their alignment. As an example of a structural quality indicator, we check that all single parts have the same length. This is done by computing the sum of the duration of all the events in streams to compare them.

*Stream issues.* At the stream level, an important property is that all the measures are correctly filled, *i.e.,* that for each measure, the total duration of the events contained corresponds to the expected measure length, according to the time signature (specified in the embedding part).

Let us also mention the problems that may occur concerning the lyrics. The association of text and music obeys some complex rules. Lyrics are decomposed into syllables. At the graphical level, syllables from the same word are linked by dashes, and melismas are indicated by underscores. People engraving music have to be aware that a correct encoding has to distinguish the syllables from the metadata that describes how they are interrelated and linked to the music. We have already found many examples where both aspects are glued together, because the engraver directly encodes continuation symbols in the text itself. As a consequence, although not directly visible, the score encoding becomes faulty: the text cannot be cleanly extracted or searched, and some notes in melismas are not properly attached to syllables. So here, quality checking involves verifying the availability of the lyrics, their encoding, and their syntax.

### 3.2 Score engraving issues

Engraving music is an art that has been refined over centuries and nowadays consists of a rich and complex mix of rules, principles, and best practices [18]. The following is a very shallow introduction to the countless issues analysis that can be envisaged at this level.

*Staves organization issues.* These issues relate to the mapping that associates a hierarchical structure (like the score content illustrated in Fig. 3, for instance) to a vertical stack of staves, each encoding one or several voices. This mapping is much more complex than a simple one-to-one association between parts and staves. For example:

- A piano part is always distributed on two staves. In many cases (but not always) they correspond to the voices played, respectively, by the left and right hands.
- Conversely, a single staff may bear two parts if saving space is important. For a large orchestra for instance, horns may be paired and one staff is allocated to each pair.

There is also a standard order for stacking parts. Instruments are grouped by family: woods are shown in the upper staves, then horns, then drums, and finally strings. In a same family, instruments are ordered with the treble ones above the others. All these conventions yield a set of quality rules that qualifies the staves' organization.

*Staff parameter issues.* This part of the taxonomy covers quality problems related to an incorrect or inconsistent assignment of parts to the staves system and on the parameters that dictate how the music content is rendered on a staff. The following is a list of examples that relate this "functional" approach to some generic quality dimensions [4].

1. *Consistency.* We check that all key signatures are consistent, including a correct transposition for transposing instruments.
2. *Correctness.* The clef should be chosen to ensure that the majority of notes lie inside the staff's range (i.e., do not show a bass part on a treble clef staff).
3. *Completeness.* We check that all parts of the score are assigned to a staff, with a maximum of two parts per staff.

*Staff layout issues.* In music theory, there are precise rules for deducing actual durations from note values and meter (TS) and common practice / recommendations for writing rhythms (using beams in particular for defining nested groups), in order to improve score readability and emphasize the meter. Again, [18] is an invaluable source on that matter.

Digital scores (*e.g.,* in MusicXML) usually contain rhythmic elements of a different nature: features related to score content, like the time signature and actual note durations, and features related to engraving content, like note symbols and beams. Despite their strong relationship, these elements can be presented independently in documents. This redundancy can be a source of inconsistency in rhythm notation.

### 3.3 Metadata issues

Metadata are attached to a music score (see Section 2.2.3). Such data provide information about the production context of the document (title, composer, provenance, data of creation, *etc.*). Metadata are particularly important for the management of the document, for instance, for classifying and retrieving data over the DSL. Quality requirements address the availability of metadata information, and their accuracy.

### 3.4 Catalog of quality requirements

During the GioQoso project, the project members exhibited about fifty data quality requirements that specifically concern music scores [12]. Table 1 contains some of them. This table is composed of four columns. For each quality requirement (i.e., for each line), the first column is an identifier, the second column contains a short description of the requirement, and the third column is the position of the requirement according to the first level of the taxonomy (Figure 7).

Let us detail some requirements of Table 1. Requirements 1 and 2 concern the completeness of the score content. Requirement 1 indicates that the parts that are expected to appear in the music score should indeed be available. Requirement 2 indicates that the available parts of the score should be aligned. Requirement 8 is the standard encoding validity constraint. For XML-based documents, it requires its conformance with the schema of the music encoding dialect (i.e., MusicXML or MEI). Requirements 17 and 18 concern the consistency of the score content. Requirement 17 indicates that the number of beats contained by a measure should respect the specified time signature. Requirement 18 is an alternative requirement, less constraining, indicating the total number of beats on contiguous frames should respect the specified time signature (in this version, some beats can "slide" from a measure to an adjacent one but the total number of beats is globally correct). Requirements 33 and 34 address the engraving of the music score. Requirements 38 to 45 concern the availability and accuracy of metadata associated with the music score.

Each quality requirement can be implemented in several ways for different purposes of the intended quality management approach. It may be implemented in order to check data and tag them where a quality problem occurs - for instance, by tagging the pieces of data that violate the requirement (we implemented an approach in the Neuma platform, as presented in Section 5). It can also be implemented in order to com-

| Id | Label | Description | Taxonomy |
|---|---|---|---|
| $R_1$ | Available parts | Each expected part appears in the music score. | Score content |
| $R_2$ | Aligned parts | The parts are aligned. | Score content |
| $R_3$ | Composer variants | The variants proposed all along the time by the composer are provided. | Score content |
| $R_6$ | No missing beat | Each measure is complete, meaning that it covers at least the number of beats defined by the time signature (if not then a note could be missing). | Score content |
| $R_7$ | Ornaments | The performance indications (appoggiaturas, slurs, articulation symbols, etc.) are uniformly present. | Score content |
| $R_8$ | Validity w.r.t. the encoding format | The music score respects the encoding format. | Score content |
| $R_{11}$ | Syntactically accurate notes | Each note is syntactically correct, meaning that both its pitch and duration lie in the accepted range. | Score content |
| $R_{17}$ | Accurate number of beats in the measure | Each measure covers exactly the number of beats defined by the time signature. | Score content |
| $R_{18}$ | Accurate number of beats w.r.t. a frame of measures | Each frame of $\mathcal{N}$ measures respects the number of beats defined in the time signature (where $\mathcal{N}$ is given as a parameter of the quality rule). More formally, for each measure $\mathcal{M}$, if the measure $\mathcal{M}$ does not strictly cover the number of beats defined in the time signature, then there is a frame of $\mathcal{N}$ adjacent measures including the frame $\mathcal{M}$ such that the number of beats of the frame is $\mathcal{N}$ times the number of beats defined by the time signature (i.e., the global frame respects the time signature). | Score content |
| $R_{19}$ | Notes in instrument tessitura | Each note of a part belongs to the tessitura of the instrument or voice that is associated with the part. | Score content |
| $R_{20}$ | Singable lyrics | Each lyric element associated with a note is singable (each unit of lyric is a syllable). | Score content |
| $R_{33}$ | Validity of the key signature | The key signature is valid. | Score engraving |
| $R_{34}$ | Validity of the clef | The clef is valid. | Score engraving |
| $R_{38}$ | Available title | The title of the music score is available. | Metadata |
| $R_{39}$ | Available composer | The composer of the music score is available. | Metadata |
| $R_{40}$ | Available date | The date of creation of the music score is available. | Metadata |
| $R_{41}$ | Available provenance | The provenance of the document (who created it and which software was used to create it) is available. | Metadata |
| $R_{45}$ | Available instruments | An instrument is associated with each part. | Metadata |

**Table 1** Some quality requirements

pute a quality score associated with a document or a corpus, by enumerating the pieces of data that satisfy the requirement. Each implementation of a quality requirement yields a *quality indicator*.

As an illustration, we consider the quality requirement $R_{11}$ "*Each note is syntactically correct, meaning that both its pitch and duration lie in the accepted range*," which expresses the need to have syntactically accurate notes. Depending on the context, such a quality requirement can lead to *tag* syntactically inaccurate notes that appear in music scores of interest. It can also lead to *compute a quality indicator*, *at the score level*, in order to assess the quality of a music score according to the requirement, like the number of syntactically correct notes over the total number of notes appearing in the score. By extension, quality indicators *at the corpus level* may easily be defined by aggregation, for instance, the average and standard deviation of the corresponding indicator at the score level, computed over the set of scores that belong to the corpus.

In the previous sections, we have defined the data model that allows the scores and the quality requirements to be modeled. We now consider how to put this framework into practice according to the usage that is made of data in the DSL.

## 4 From usage to quality: fitness for use

Music scores are being produced by individuals and institutions with highly variable motivations and skills. By "motivation" we denote here the purpose of creating and editing a score in digital format. A first motivation is obviously the production of material for performers, with various levels of demands. Some users may content themselves with schematic notation of simple songs, whereas others will aim at professional editing with high quality standards. The focus here is on rendering, readability, and manageability of the score sheets in performance situations. Another category of users (with, probably, some overlap) is scientific editors, whose purpose is rather an accurate and long-term preservation of the source content (including variants and composer's annotations). The focus will be put on completeness: all variants are represented, editor's corrections are fully documented, links are provided to other resources if relevant, and collections are constrained by carefully crafted editorial rules. Overall, the quality of such projects is estimated by the ability of a document to convey the composer's intent as respectfully as possible as it can be perceived through the available sources. Librarians are particularly interested in the searchability of their collections, with rich annotations linked to taxonomies [28]. We finally mention analysts, teachers, and musicologists: their focus is put on the core music material, and less on rendering concerns. In such a context, a part of the content may be missing without harm; accuracy, accessibility, and clarity of the features investigated by the analytic process are the main quality factors.

So even if a lot of indicators may be considered for assessing the quality of music scores, not all of them may be used for evaluating data quality in a given operational context. An important property concerning data quality is that it is defined by its *fitness for use* of data [26,4], meaning that the quality assessment involves dimensions and indicators that are relevant to a given (set of) user(s) for a given *usage*. User $u_1$ may require some quality indicators for a specific usage, and some other indicators for another one, which can be completely different than those needed by user $u_2$.

*Example 2* Let us consider Table 1 again. Let us also consider four users of a given DSL:

– Maria, a music performer, who retrieves music scores in order to play music with her jazz band on Saturday night;
– Cecile, a music analyst, who searches for similar patterns in the parts of a music score by using an automatic tool based on an algorithm that analyzes each score, measure by measure;
– David, a musicologist, who conducts a philological study on the sources – and all their variants – of a composer;
– Alan, a librarian, who manages a DSL, providing access (searching and retrieving) to collections of music scores.

These users consider the same music scores, stemming from the same DSL. But they obviously have different usages of the data, leading to different quality requirements. For Maria (the music performer), the quality requirements mainly concern the rendering of the music score, which covers the completeness over the performance-related information. Most of the metadata is of no particular interest for her. For Cecile (the analyst) who executes an algorithm that automatically analyzes the score measure by measure, the requirements are not for the rendering but the respect of the encoding and its strict consistency throughout the score. For David (the musicologist), tracing the composer's work through the source's variants is the key, and he cares above all about the completeness of the source's encoding, and on the metadata that describes the provenance of each fragment. For his part, Alan (the librarian) is particularly interested in the searchability of his collections, with rich annotations of metadata. His primary objective is to be able to provide retrievable music scores, their form (searchable content or not) being only Alan's secondary purpose. Table 2 indicates the importance that each of these users would assign to the quality requirements given in Table 1,[4] illustrating the *fitness for use* aspect of data quality in a multi-user system, leading to a subjective definition of quality that is specific to each (group of) user(s).

In practice, the first problem of quality management is to elicit the quality requirements, for each user, with regard to her/his usage of data. This is a methodological issue.

*Eliciting quality requirements.* Eliciting data quality requirements means choosing a set of quality indicators, and possibly thresholds associated with them, that allows us to measure how the data fit the quality requirements *according to a given data usage*. This is a delicate task for which dedicated methodological guidelines have been proposed in literature. The well-recognized method *Goal Question Metric* paradigm [3] suggests defining quality requirements according to a top-down analysis, going from the business goal to its corresponding quality indicators. We illustrate its main stages in our context.

*GQM – Stage 1.* For each user (or each user role) and for each of his/her usage of data, conceptual *business goals* are identified. A business goal specifies the intent of a quality measurement according to some data usage.

*Example 3* Assume that the business user Cecile retrieves the music scores of a DSL in order to *($\mathcal{G}$) Perform a given algorithm that searches for similar patterns in the parts of a music score.* This is a business goal.

[4] Of course one can discuss this assignment according to the context as it only reflects a general trend of such users' visions according to their roles.

| Quality requirement Id | Label | Maria (Performer) | Cecile (Analyst) | David (Musicologist) | Alan (Librarian) |
|---|---|---|---|---|---|
| $R_1$ | Available parts | ++ | + | + | - |
| $R_2$ | Aligned parts | ++ | - | ++ | |
| $R_3$ | Composer variants | - | + | ++ | + |
| $R_6$ | No missing beat | ++ | + | - | - |
| $R_7$ | Ornaments | ++ | - | ++ | - |
| $R_8$ | Validity w.r.t. the encoding format | - | ++ | | ++ |
| $R_{17}$ | Accurate number of beats in the measure | - | ++ | - | - |
| $R_{18}$ | Accurate number of beats w.r.t. a frame of measures | ++ | - | ++ | - |
| $R_{19}$ | Notes in instrument tessitura | ++ | - | ++ | - |
| $R_{27}$ | Singable lyrics | ++ | - | + | - |
| $R_{33}$ | Validity of the key signature | ++ | - | ++ | - |
| $R_{34}$ | Validity of the clef | ++ | + | ++ | - |
| $R_{38}$ | Available title | + | - | + | ++ |
| $R_{39}$ | Available composer | + | - | + | ++ |
| $R_{40}$ | Available date | - | - | ++ | ++ |
| $R_{41}$ | Available provenance | - | - | + | ++ |
| $R_{45}$ | Available instruments | ++ | - | + | ++ |

Legend:
- depicts a minor requirement
+ depicts a significant requirement
++ depicts an important requirement

**Table 2** Some quality concerns according to usage

*GQM – Stage 2.* Each goal is refined into a set of operational *quality questions*, which are a first step toward eliciting the quality requirements.

*Example 3* (Continuation) Cecile may express that the results of her study are relevant provided that data (input of the algorithm) is complete enough. She also indicates that the algorithm computes relevant results provided that data is accurate enough. Quality questions associated with this use case could then be the following ones.

*(QQ1) Does the data contain all the needed information?*
*(QQ2) Are the notes accurate?*

*GQM – Stage 3.* Each quality question is itself expressed in terms of a set of quantifiable quality requirements. In our context of DSL data, these requirements are taken from the data quality model introduced in Section 3.

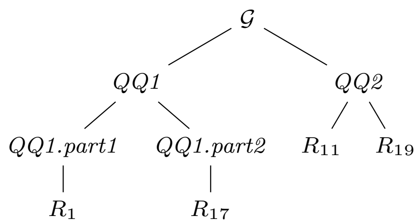


**Fig. 8** Eliciting quality requirements (example)

*Example 3* (Continuation) Considering our example, the quality question *(QQ1)* could be refined by Cecile into two more specific quality requirements.

A first quantitative quality question associated with *(QQ1)* could be *(QQ1.part1) Are the parts available?*, corresponding to the quality requirement $R_1$ of Table 1.

A second quantitative quality question associated with *(QQ1)* could be *(QQ1.part2) Does each measure cover the expected number of beats?*, corresponding to the quality requirement $R_{17}$ of Table 1.

Concerning the quality question *(QQ2)*, it could be refined into a quality indicator that measures the syntactic accuracy of the notes, meaning that each note should be an existing one (which belongs to the usual range of notes), and that the note belongs to the tessitura of its instrument, corresponding to the quality requirements $R_{11}$ and $R_{19}$ of Table 1.

So, for the example, the quality requirements $R_1$, $R_{17}$, $R_{11}$ and $R_{19}$ of Table 1 concern the usage *($\mathcal{G}$) Perform a given algorithm that searches for similar patterns in the parts of a music score* of $u_{analyst}$, derived as illustrated in Figure 8 by following the GQM methodology.

*Implementing personalization.* In practice, the *fitness for use* inherent feature of data quality implies that a DSL has to offer a personalization of the data quality information, for instance, by taking users' profiles [11] into account (a general definition of a profile in the context of DSL is proposed in [6]). In terms of implementation, the DSL allows the user to interact with its system via graphical user interface tools (GUI), to let her/him define her/his profile. Implementing such a feature goes from proposing simple check boxes for filtering data, to managing stored and possibly pre-defined profiles associated with registered users.

It should be noted that sheet music scores exist that, in the original version (i.e., as written by the author), do not adhere to the quality principles that we have proposed above. This is the case of *rubato* sections, where the duration of measures does not reflect the information encoded in the time signature. These situations are not the main target of our work, since they are not created by a wrong digital encoding of the sheet music. However, they can still disrupt the workflow of automatic systems and it is useful to automatically detect them. The choice of whether to accept them or to treat those excerpts differently will, again, depend on the specific user application.

## 5 Implementation issues

The quality framework is embedded in the Neuma platform. Neuma [27,23] is a DSL devoted to the preservation and the dissemination of symbolic music content (scores). It is open to musicologists, musicians, and music publishers. It consists of a repository dedicated to the storage of large collections of digital scores, where users/applications can upload their documents. The corpora of Neuma are publicly available, in open access, at `http://neuma.huma-num.fr`.

The conceptual quality management framework presented in the previous sections is implemented in the form of a Neuma module that is denoted by the name of the project GioQoso. The GioQoso quality module [5] is integrated in the Neuma library, but it is an independent web service component that can be used to analyze any XML score accessible at a public URL. The service is publicly available at `http://neuma.huma-num.fr/quality`.

We now consider the implementation of the GioQoso quality module in Neuma. The architecture of Neuma consists of several layers (see Figure 9): a *storage layer* manages the persistent storage and access paths; the *models layer* organizes the information in high-level structures that support the logic of the system (see Section 2); the *functional layer* provides the implementation of the web services offering functionalities of Neuma (the GioQoso quality module is one of them); finally, the *presentation layer* offers the GUI interface that allows a user to interact with the system, and provides the entry point to the web services.

In order to access data, the quality module interfaces with the models layer, which itself interacts with the storage layer where data, metadata, and quality annotations are stored. The quality module also interfaces with the GUI, in order to display quality information to the users.

*Database.* At the storage layer, music scores are stored as XML documents structured according to the MEI specification. MusicXML documents can be imported as well, but in that case an internal conversion is done first to obtain a MEI encoding. The MEI features provide two major advantages in our context.

First, each element of the score (notes, rests, slurs, measures, staves, *etc.*) has a unique Id. This is essential to *annotate* this element with a semantic label, in our case, a quality indicator that indicates the violation of a quality requirement. For instance, a note can be annotated with a *missing lyrics* indicator, or a measure with an *incomplete duration* indicator.

A second advantage of the MEI encoding is that it comes with several analyses and interactive tools. We use the Verovio toolkit [25] in particular to display and interact with the score. Verovio relies on a conversion from MEI to SVG that preserves the Id of elements. As a result, an annotation (*i.e.*, some meaning attached to a note or a measure) can be graphically displayed as a decoration of the corresponding SVG element.

The ability to play a MIDI rendering of a score, possibly starting from any note, is also a Verovio feature. This functionality corresponds to the standard “Play” option offered by all score engravers, and is quite a useful tool when it comes to checking the content of a score.

*Quality management.* The quality of the documents is analyzed on-the-fly in order to complete them with quality annotations. Each annotation is an instance of a quality indicator, and the indicators themselves are organized as a forest, based on the taxonomy presented in Section 3.

The computation of the indicators is based on procedures that involve

- a parsing of the MEI documents encoding the score, for instance, to extract beaming trees on which duration properties can be checked, or
- calling routines of music21 [10], for instance, to extract the event durations.

Details of such procedures can be found in [9], [13] and [5].

The taxonomy of the quality model is extensible. We add new rules regularly, based on input from our scientific experts (the CESR and IReMus musicology labs), on best notational practices found in reference sources on score rendering/engraving, *e.g.,* [18], and on

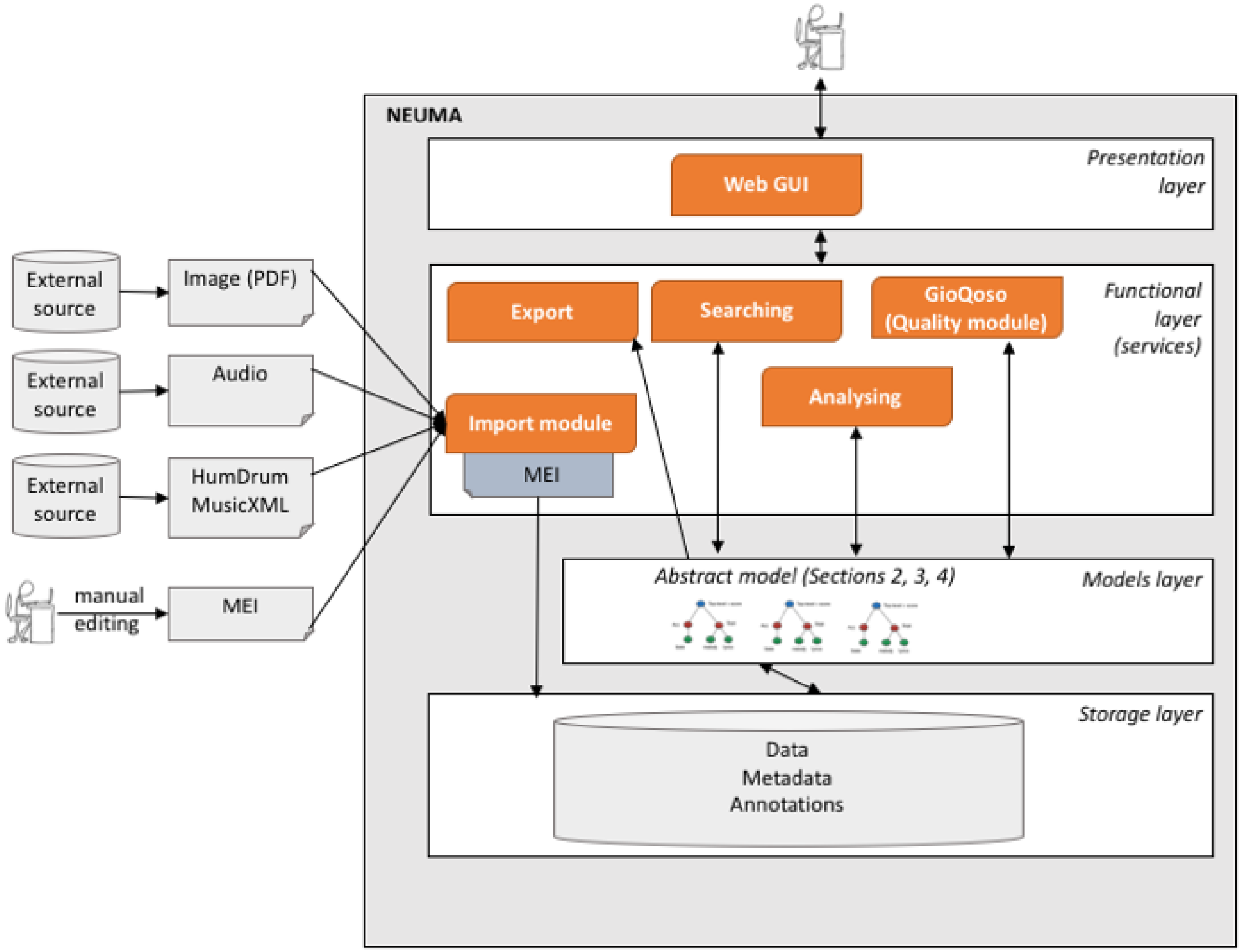


**Fig. 9** Architecture of NEUMA

mere exploration of various online score libraries that reveal many encoding and rendering issues.

Figure 10 shows the current status of the GUI of the GioQoso quality tool. In the interface, the indicators are displayed in the top-right part of the user interface (the column "quality concepts" in Figure 10). Each indicator comes with a description that can be highlighted by dragging the mouse over its name. Every annotation is displayed as a small colored circle above the elements or groups of elements that constitute the annotated fragment. Its color characterizes a specific quality indicator.

The user can hide/show a set of annotations by clicking on any level of the model tree. This makes it convenient to focus on a particular aspect, or to ignore some indicators altogether if they are deemed irrelevant (this is a simple implementation of the user profile notions, which can easily be extended to consider more complex profiles, if needed).

*Interactions.* Actions can be carried out by the user. Each annotation can be inspected in detail by clicking on it. The *Info box* part of the interface then displays details on the related score elements, and on their annotations (there might be many). A form is also provided to report an annotation error, or to complete existing annotations. Such inputs might become quite useful in the future to include user feedback in the context of a large collaborative system.

Note that since the score is loaded from its remote location, the user can correct the identified issue on her local version directly. It suffices then to reload GioQoso to trigger a new evaluation of the quality requirements that will hopefully show that some formerly-identified quality issues have been fixed. GioQoso can therefore be seen as a complementary tool that is closely and easily integrated to the user's score production environment. The only requirement is for the score under production to be accessible at a fixed URL.

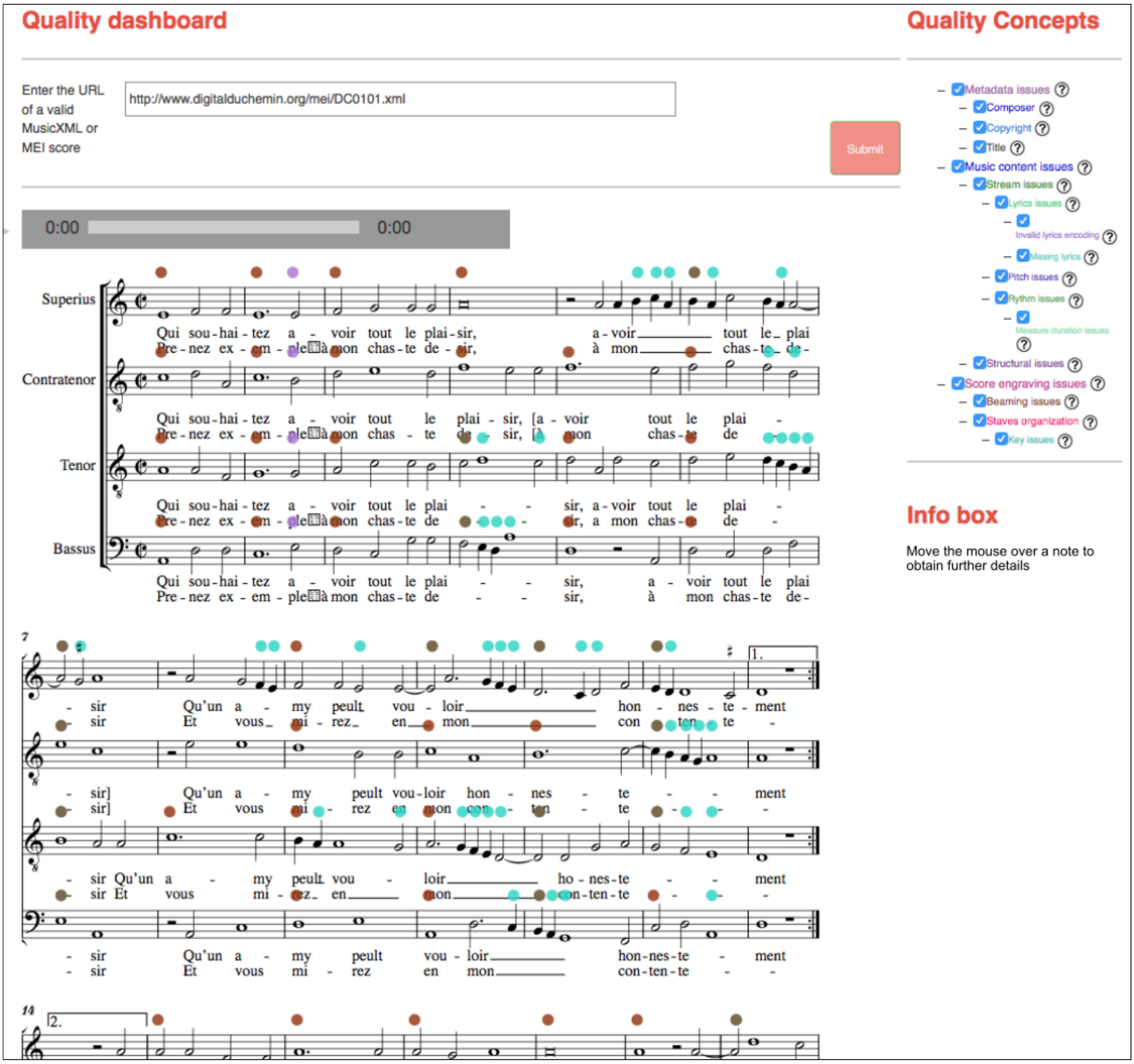


**Fig. 10** The GioQoso User Interface

## 6 Conclusion and perspectives

In this article, we presented a framework for assessing data quality in a digital score library. The whole framework is composed of several contributions that were proposed during the multidisciplinary GioQoso project (2016-2019), which gathered musicologists and computer scientists to study the problem of managing data quality in digital score libraries.

First, we introduced a data model that allows the content of a music score to be modeled. This model is based on the music score production process, leading us to distinguish the score content from the engraving issues. We also presented the data quality model that defines quality requirements according to a DSL-specific taxonomy that classifies them. We then explained the aspect of *fitness for use* of data quality, which is crucial in practice as data quality assessment involves requirements that are relevant according to each use case. Finally, an implementation of this framework concretely illustrated the approach, in the form of the GioQoso web service embedded in the NEUMA digital score library.

The GioQoso project was a first step toward quality management of digital score libraries. It opens a lot of research perspectives. We propose some of them below.

*Quality assessment.* Several of the indicators identified during our preliminary study cannot be evaluated from the notation itself but require an external reference. This is the case for the indicators that concern the *semantic accuracy* of the information, which checks that the provided value accurately models its real world value, known as being a difficult issue. We illustrate this notion on the simple example of the *year of birth* of a composer, embedded in the metadata of a music score. Checking the availability of the expected information is rather easy. Typically, this can be done by parsing the score document, or more specifically, the parts of the document where the information should appear (for instance, looking for the expected tags in an XML-based format). But even if data is available, this does not mean that the information provided is accurate. Checking the *syntactic accuracy*, which means checking that the data respects the expected format of the information, is feasible. For instance, one can check that a year has the expected syntax of a date (a number, in an expected range of dates). But even if the year of birth respects the expected format, this still does not mean that it is *semantically accurate*. Checking the semantic accuracy requires external references.

Several solutions can be envisioned in order to provide external references. A first approach is the **collaborative** evaluation (some methodologies were proposed e.g., in [14,2]), such as the one based on crowdsourcing, in which users themselves tag the quality problems in the document. Another approach consists in exploiting open **semantic web data** by interlinking the DSL collections with other data sources [30].

*Quality improvement.* A second important perspective is to address another aspect of quality management, namely **quality improvement** techniques [4], in order to fix the detected quality problem. Such an improvement can be fully automatic in some specific cases (e.g., filling incomplete measures with rests) but in general, the goal is to help users to identify the insert/update process deficiencies, and to suggest effective improvement strategies.

**Acknowledgements** This work has been partly funded by the French National Center for Scientific Research (CNRS) under the *défi Mastodons* GioQoso.

## References

1. BnF's digital library Gallica. `http://gallica.bnf.fr`
2. Acosta, M., Zaveri, A., Simperl, E., Kontokostas, D., Auer, S., Lehmann, J.: Crowdsourcing Linked Data Quality Assessment. In: Proceedings of the International Semantic Web Conference (ISWC'13), pp. 260–276 (2013)
3. Basili, V.R., Caldiera, G., Rombach, H.D.: Encyclopedia of Software Engineering, chap. The Goal Question Metric Approach. Wiley (1994)
4. Batini, C., Scannapieco, M.: Data Quality: Concepts, Methodologies and Techniques. Data-Centric Systems and Applications. Springer (2016)
5. Besson, V., Fiala, D., Rigaux, P., Thion, V.: Gioqoso, an Online Quality Evaluation Tool for MEI Scores. Music Encoding Conference (MEC'18) (2018). URL `https://hal.archives-ouvertes.fr/hal-01708859`. Poster
6. Besson, V., Gurrieri, M., Rigaux, P., Tacaille, A., Thion, V.: A Methodology for Quality Assessment in Collaborative Score Libraries. In: Proceedings of the International Society for Music Information Retrieval Conference (ISMIR'16) (2016). URL `https://hal.inria.fr/hal-01316014`
7. Cambouropoulos, E.: Voice and stream: Perceptual and computational modeling of voice separation. Music Perception: An Interdisciplinary Journal **26**(1), 75–94 (2008)
8. Cherfi, S., Guillotel, C., Hamdi, F., Rigaux, P., Travers, N.: Ontology-Based Annotation of Music Scores. In: Intl. Conf. on Knowledge Capture (K-CAP'17) (2017). Austin, Texas, Dec. 4-6 2017
9. Cherfi, S., Hamdi, F., Rigaux, P., Thion, V., Travers, N.: Formalizing Quality Rules on Music Notation – an Ontology-Based Approach. In: Intl. Conf. on Technologies for Music Notation and Representation (TENOR'17) (2017)
10. Cuthbert, M.S., Ariza, C.: music21: A toolkit for computer-aided musicology and symbolic music data. In: Proceedings of the International Society for Music Information Retrieval (ISMIR'10) (2010)
11. Eke, C., Norman, A., Shuib, L., Nweke, H.: A survey of user profiling: State-of-the-art, challenges and solutions. IEEE Access **PP**, 1–1 (2019). DOI 10.1109/ACCESS.2019.2944243
12. Fiala, D., Rigaux, P., Tacaille, A., Thion, V., the members of The Gioqoso project: Data Quality Rules for Digital Score Libraries. Research report, IRISA, Université de Rennes (2018). URL `https://hal.inria.fr/hal-01734821`
13. Foscarin, F., Fiala, D., Jacquemard, F., Rigaux, P., Thion, V.: Gioqoso, an online Quality Assessment Tool for Music Notation. Proceedings of the International Conference on Technologies for Music Notation and Representation (TENOR'18) (2018). URL `https://hal.inria.fr/hal-01895171`. Poster
14. Gil, Y., Ratnakar, V.: Trusting information sources one citizen at a time. In: Proceedings of the International Semantic Web Conference (ISWC'02), pp. 162–176 (2002)
15. Gioqoso project web site (consulted in 2020)
16. Global Chant Database. `http://www.globalchant.org/`
17. Good, M.: The Virtual Score: Representation, Retrieval, Restoration, chap. "MusicXML for Notation and Analysis", pp. 113–124. W. B. Hewlett and E. Selfridge-Field, MIT Press (2001)
18. Gould, E.: Behind Bars. Faber Music (2011)
19. International Music Score Library Project or Petrucci Music Library (IMSLP). `https://imslp.org`

20. The lost voices project. http://www.digitalduchemin.org
21. Music Encoding Initiative. http://www.music-encoding.org (2015). Accessed April 2019
22. Music Notation Community Group. https://www.w3.org/community/music-notation/ (2018). Accessed April 2019
23. NEUMA. http://neuma.huma-num.fr
24. The Open Score Project (consulted in 2020). https://www.open-score.fr
25. Pugin, L., Zitellini, R., Roland, P.: Verovio: A library for Engraving MEI Music Notation into SVG. In: Proceedings of the International Society for Music Information Retrieval (ISMIR'14), pp. 107–112 (2014). http://verovio.org
26. Redman, T.C.: Data Quality for the Information Age. Artech House Inc. (1996)
27. Rigaux, P., Abrouk, L., Audéon, H., Cullot, N., Davy-Rigaux, C., Faget, Z., Gavinet, E., Gross-Amblard, D., Tacaille, A., Thion, V.: The Design and Implementation of NEUMA, a Collaborative Digital Score Library - Requirements, architecture, and models. Intl. Journal On Digital Libraries (JODL) pp. 1–24 (2012). https://hal.archives-ouvertes.fr/hal-01126064
28. Riley, J., Mayer, C.A.: Ask a Librarian: The Role of Librarians in the Music Information Retrieval. In: Proceeding of International Conference on Music Information Retrieval (ISMIR) (2006)
29. Rolland, P.: The Music Encoding Initiative (MEI). In: Proceedings of the International Conference on Musical Applications Using XML, pp. 55–59 (2002)
30. Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auer, S.: Quality assessment for linked data: A survey. Semantic Web **7**(1), 63–93 (2016). DOI 10.3233/SW-150175. URL http://dx.doi.org/10.3233/SW-150175